Amendments to the Claims

Claim 1 (original): A plurality of arrays representing a structured document in an array-based storage format, wherein the arrays reside on one or more computer-readable media and comprise:

an element name array, the element name array comprising an element name entry for each element in the structured document;

an element value array, the element value array comprising an element value entry for each element in the structured document;

an attribute array, the attribute array comprising an attribute entry for each element in the structured document;

a parent array, the parent array comprising a parent entry for each element in the structured document and wherein a value of each parent entry identifies a parent of the element; and

a child array, the child array comprising a child entry for each element in the structured document and wherein a value of each child entry identifies zero or more children of the element.

Claim 2 (original): The arrays according to Claim 1, wherein each element name entry specifies a starting name position and a name length.

Claim 3 (original): The arrays according to Claim 2, wherein the starting name position is relative to a beginning of a storage buffer wherein a name of each of the elements is stored.

Claim 4 (original): The arrays according to Claim 1, wherein each element name entry specifies a

starting name position and an ending name position, and wherein the starting and ending name positions are relative to a beginning of a storage buffer wherein a name of each of the elements is stored.

Claim 5 (original): The arrays according to Claim 1, wherein each element value entry specifies a starting value position and a value length.

Claim 6 (original): The arrays according to Claim 5, wherein the starting value position is relative to a beginning of a storage buffer wherein a value of each of the elements is stored.

Claim 7 (currently amended): The arrays according to Claim [[5]] 1, wherein each element value entry specifies a starting value position and an ending value position, and wherein the starting and ending value positions are relative to a beginning of a storage buffer wherein a value of each of the elements is stored.

Claim 8 (original): The arrays according to Claim 1, wherein each attribute entry specifies a reference to a secondary array, wherein the secondary array comprises a secondary attribute entry for each of one or more attributes of those ones of the elements which have attributes, and a null value otherwise.

Claim 9 (original): The arrays according to Claim 8, wherein each secondary attribute entry specifies a starting name position and a length for a name of the attribute, and a starting value

position and a length for a value of the attribute.

Claim 10 (original): The arrays according to Claim 8, wherein each secondary attribute entry specifies a starting name position and an ending name position for a name of the attribute, and a starting value position and an ending value position for a value of the attribute.

Claim 11 (original): A plurality of arrays representing a structured document in an array-based storage format, wherein the arrays reside on one or more computer-readable media and comprise:

an element name array, the element name array comprising an element name entry for each element in the structured document, wherein each element name entry specifies a starting name position and one of (1) a name length or (2) an ending name position;

an element value array, the element value array comprising an element value entry for each element in the structured document, wherein each element value entry specifies a starting value position and one of (1) a value length or (2) an ending value position;

a parent array, the parent array comprising a parent entry for each element in the structured document and wherein a value of each parent entry identifies a parent of the element; and

a child array, the child array comprising a child entry for each element in the structured document and wherein a value of each child entry identifies zero or more children of the element.

Claim 12 (original): A computer program product embodied on one or more computer-readable media, the computer program product adapted for representing a source document encoded in an

extensible structured notation using a plurality of arrays and comprising:

computer-readable program code means for generating an element name array, the element name array comprising an element name entry for each element in the source document, wherein each element name entry specifies a starting name position and one of (1) a name length or (2) an ending name position;

computer-readable program code means for generating an element value array, the element value array comprising an element value entry for each element in the source document, wherein each element value entry specifies a starting value position and one of (1) a value length or (2) an ending value position;

computer-readable program code means for generating a parent array, the parent array comprising a parent entry for each element in the source document and wherein a value of each parent entry identifies a parent of the element;

computer-readable program code means for generating a child array, the child array comprising a child entry for each element in the source document and wherein a value of each child entry identifies zero or more children of the element; and

computer-readable program code means for storing the generated arrays in memory or writing the generated arrays to one or more storage media.

Claim 13 (original): The computer program product according to Claim 12, further comprising:

computer-readable program code means for generating an attribute array, the attribute array comprising an attribute entry for each element in the structured document, wherein each attribute entry specifies a reference to a secondary array and wherein the secondary array

comprises a secondary attribute entry for each of one or more attributes of those ones of the elements which have attributes, and a null value otherwise; and wherein each secondary attribute entry specifies a starting name position and one or (1) an ending name position or (2) a length for a name of the attribute, and a starting value position and one or (1) an ending value position or (2) a length for a value of the attribute.

Claim 14 (original): The computer program product according to Claim 12, wherein the extensible structured notation is XML (Extensible Markup Language).

Claim 15 (original): The computer program product according to Claim 12, further comprising computer-readable program code means for generating an output structured document from the arrays.

Claim 16 (original): A computer program product embodied on one or more computer-readable media, the computer program product adapted for creating a plurality of arrays to represent a source document encoded in a machine-oriented extensible structured notation ("mXML") and comprising:

computer-readable program code means for obtaining a node count from the source document;

computer-readable program code means for generating the arrays based on the node count; and

computer-readable program code means for processing a plurality of node specifications

from the source document, further comprising:

computer-readable program code means for obtaining an element name specification from the node specification;

computer-readable program code means for storing element name information in an element name array, using the element name specification;

computer-readable program code means for obtaining an attribute list specification from the node specification;

computer-readable program code means for storing attribute information in an attribute array, using the attribute list specification;

computer-readable program code means for obtaining a child list specification from the node specification;

computer-readable program code means for storing child information in a child array, using the child list specification;

computer-readable program code means for storing parent information in a parent array, using the child list specification;

computer-readable program code means for obtaining an element value specification from the node specification; and

computer-readable program code means for storing element value information in an element value array, using the element specification.

Claim 17 (original): The computer program product according to Claim 16, further comprising computer-readable program code means for generating an output mXML document by traversing

the plurality of arrays.

Claim 18 (currently amended): A computer program product embodied on one or more computer-readable media, the computer program product adapted for ~~efficiently~~ transforming a structured document and comprising:

computer-readable program code means for creating an array-based representation of the structured document, further comprising:

computer-readable program code means for creating an element name array to store information pertaining to a name of each of a plurality of elements in the structured document;

computer-readable program code means for creating an element value array to store information pertaining to a value of each of the elements;

computer-readable program code means for creating an attribute array to store information pertaining to a name and a value of each of zero or more attributes of each of the elements;

computer-readable program code means for creating a parent array to store information pertaining to a parent of each of the elements; and

computer-readable program code means for creating a child array to store information pertaining to zero or more children of each of the elements;

computer-readable program code means for obtaining an identification of a particular element of the structured document which is to be transformed;

computer-readable program code means for locating an entry for the particular element in

the arrays-based representation; and

computer-readable program code means for transforming information represented by the located entry.

Claim 19 (original): The computer program product according to Claim 18, wherein the identification is an element name and wherein the computer-readable program code means for locating further comprises computer-readable program code means for searching the element name array to find a match with the identification.

Claim 20 (original): The computer program product according to Claim 18, wherein the identification is an ordinal representing a relative position of the particular element in the structured document, and wherein the computer-readable program code means for locating further comprises computer-readable program code means for using the ordinal as an index to access one or more of the arrays in the array-based representation.

Claim 21 (original): A system for representing a source document encoded in an extensible structured notation using a plurality of arrays, comprising:

means for generating an element name array, the element name array comprising an element name entry for each element in the source document, wherein each element name entry specifies a starting name position and one of (1) a name length or (2) an ending name position;

means for generating an element value array, the element value array comprising an element value entry for each element in the source document, wherein each element value entry

specifies a starting value position and one of (1) a value length or (2) an ending value position;

means for generating a parent array, the parent array comprising a parent entry for each element in the source document and wherein a value of each parent entry identifies a parent of the element;

means for generating a child array, the child array comprising a child entry for each element in the source document and wherein a value of each child entry identifies zero or more children of the element; and

means for storing the generated arrays in memory or writing the generated arrays to one or more storage media.

Claim 22 (original): The system according to Claim 21, further comprising:

means for generating an attribute array, the attribute array comprising an attribute entry for each element in the structured document, wherein each attribute entry specifies a reference to a secondary array and wherein the secondary array comprises a secondary attribute entry for each of one or more attributes of those ones of the elements which have attributes, and a null value otherwise; and wherein each secondary attribute entry specifies a starting name position and one or (1) an ending name position or (2) a length for a name of the attribute, and a starting value position and one or (1) an ending value position or (2) a length for a value of the attribute.

Claim 23 (original): The system according to Claim 21, wherein the extensible structured notation is XML (Extensible Markup Language).

Claim 24 (original): The system according to Claim 21, further comprising means for generating an output structured document from the arrays.

Claim 25 (original): A system for creating a plurality of arrays to represent a source document encoded in a machine-oriented extensible structured notation ("mXML"), comprising:

means for obtaining a node count from the source document;

means for generating the arrays based on the node count; and

means for processing a plurality of node specifications from the source document, further comprising:

means for obtaining an element name specification from the node specification;

means for storing element name information in an element name array, using the element name specification;

means for obtaining an attribute list specification from the node specification;

means for storing attribute information in an attribute array, using the attribute list specification;

means for obtaining a child list specification from the node specification;

means for storing child information in a child array, using the child list specification;

means for storing parent information in a parent array, using the child list specification;

means for obtaining an element value specification from the node specification; and

means for storing element value information in an element value array, using the

element specification.

Claim 26 (original): The system according to Claim 25, further comprising means for generating an output mXML document by traversing the plurality of arrays.

Claim 27 (currently amended): A system for ~~efficiently~~ transforming a structured document, comprising:

means for creating an array-based representation of the structured document, further comprising:

means for creating an element name array to store information pertaining to a name of each of a plurality of elements in the structured document;

means for creating an element value array to store information pertaining to a value of each of the elements;

means for creating an attribute array to store information pertaining to a name and a value of each of zero or more attributes of each of the elements;

means for creating a parent array to store information pertaining to a parent of each of the elements; and

means for creating a child array to store information pertaining to zero or more children of each of the elements;

means for obtaining an identification of a particular element of the structured document which is to be transformed;

means for locating an entry for the particular element in the arrays-based representation;

and

means for transforming information represented by the located entry.

Claim 28 (original): The system according to Claim 27, wherein the identification is an element name and wherein the means for locating further comprises means for searching the element name array to find a match with the identification.

Claim 29 (original): The system according to Claim 27, wherein the identification is an ordinal representing a relative position of the particular element in the structured document, and wherein the means for locating further comprises means for using the ordinal as an index to access one or more of the arrays in the array-based representation.

Claim 30 (original): A method for representing a source document encoded in an extensible structured notation using a plurality of arrays, comprising the steps of:

generating an element name array, the element name array comprising an element name entry for each element in the source document, wherein each element name entry specifies a starting name position and one of (1) a name length or (2) an ending name position;

generating an element value array, the element value array comprising an element value entry for each element in the source document, wherein each element value entry specifies a starting value position and one of (1) a value length or (2) an ending value position;

generating a parent array, the parent array comprising a parent entry for each element in the source document and wherein a value of each parent entry identifies a parent of the element;

generating a child array, the child array comprising a child entry for each element in the source document and wherein a value of each child entry identifies zero or more children of the element; and

storing the generated arrays in memory or writing the generated arrays to one or more storage media.

Claim 31 (currently amended): The method according to Claim [[21]] 30, further comprising the steps of:

generating an attribute array, the attribute array comprising an attribute entry for each element in the structured document, wherein each attribute entry specifies a reference to a secondary array and wherein the secondary array comprises a secondary attribute entry for each of one or more attributes of those ones of the elements which have attributes, and a null value otherwise; and wherein each secondary attribute entry specifies a starting name position and one or (1) an ending name position or (2) a length for a name of the attribute, and a starting value position and one or (1) an ending value position or (2) a length for a value of the attribute.

Claim 32 (original): The method according to Claim 30, wherein the extensible structured notation is XML (Extensible Markup Language).

Claim 33 (original): The method according to Claim 30, further comprising the step of generating an output structured document from the arrays.

Claim 34 (original): A method for creating a plurality of arrays to represent a source document encoded in a machine-oriented extensible structured notation ("mXML"), comprising the steps of:

obtaining a node count from the source document;

generating the arrays based on the node count; and

processing a plurality of node specifications from the source document, further comprising the steps of:

obtaining an element name specification from the node specification;

storing element name information in an element name array, using the element name specification;

obtaining an attribute list specification from the node specification;

storing attribute information in an attribute array, using the attribute list specification;

obtaining a child list specification from the node specification;

storing child information in a child array, using the child list specification;

storing parent information in a parent array, using the child list specification;

obtaining an element value specification from the node specification; and

storing element value information in an element value array, using the element specification.

Claim 35 (original): The method according to Claim 34, further comprising the step of generating an output mXML document by traversing the plurality of arrays.

Claim 36 (currently amended): A method for ~~efficiently~~ transforming a structured document, comprising the steps of:

creating an array-based representation of the structured document, further comprising the steps of:

creating an element name array to store information pertaining to a name of each of a plurality of elements in the structured document;

creating an element value array to store information pertaining to a value of each of the elements;

creating an attribute array to store information pertaining to a name and a value of each of zero or more attributes of each of the elements;

creating a parent array to store information pertaining to a parent of each of the elements; and

creating a child array to store information pertaining to zero or more children of each of the elements;

obtaining an identification of a particular element of the structured document which is to be transformed;

locating an entry for the particular element in the arrays-based representation; and

transforming information represented by the located entry.

Claim 37 (original): The method according to Claim 36, wherein the identification is an element name and wherein the locating step further comprises the step of searching the element name array to find a match with the identification.

Claim 38 (original): The method according to Claim 36, wherein the identification is an ordinal representing a relative position of the particular element in the structured document, and wherein the locating step further comprises the step of using the ordinal as an index to access one or more of the arrays in the array-based representation.